

© 2010 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

# A New Finger Inverse Kinematics Method for an Anthropomorphic Hand



Choukri Bensalah, Mohamed Abderrahim and Juan Gonzalez Gomez

**Abstract**—In this paper, a new method for solving the inverse kinematics of the fingers of an anthropomorphic hand is proposed. Our approach combines a Modified Selectively Damped Least Squares (MSDLS) and Jacobian Transpose (JT) methods. The main advantages of this method with respect to the ordinary SDLS are: optimal Cartesian increment, shorter computation time and better response near singularity configurations. The original JT method exhibits a strong shattering with small magnitudes which occurs near the goal position or in the case of unreachable positions. Like in the SDLS, a damping factor was applied to each input singular vector to filter the undesirable behavior. A comparative study between the MSDLS applied to the inverse Jacobian and JT matrix is developed to investigate manipulator performance in critical end-point positions of the index finger of a commercial *anthropomorphic robotic hand* and also to evaluate the impact of the increment length on computation time.

## I. INTRODUCTION

Achieving higher autonomy and dexterity for robotic hands in the manipulation of objects and tools are among the major challenges to be resolved. The fingers of a robotic hand can be viewed as manipulators which are required to be controlled as a single set to achieve a given task. Performing the dextrous tasks requires an efficient control philosophy in both high and low level control. The fingertip position control is considered as one of those low level control schemes that achieve the contact position with the object using the inverse kinematics model. Like in humans skills, the soft-contacts are defined as a small area rather than a point [1]. Using this kind of contact, the required inverse kinematics scheme does not need to be too accurate. However, the inverse kinematics scheme must be fast enough to allow the real time implementation.

During the late 80's and early 90's, the calculation of the inverse kinematics of manipulators has been widely studied. These studies provide several theoretical concepts such as the ill-conditioned Jacobian matrix. Physically, this means that the manipulator is at singularity configuration or in its neighborhood. In this situation, a small deviation in the cartesian space provides a high change in joint space variables which is not desirable. It is extremely important to ensure a good behavior of the manipulator in this critical configuration.

This work was partially supported by the European project HANDLE, FP7-231640. Choukri Bensalah is supported by the Spanish ministry MICINN through FPI scholarship within the project DPI-2005-04302

C. Bensalah, M. Abderrahim and J.G. Gomez are with the Roboticslab, Systems Engineering and Automation Department, University Carlos III of Madrid, Spain (emails: {cbensala;mohamed;jggomez}@ing.uc3m.es)

Many researchers have proposed methods to avoid singularities. The most successful is the Damped Least Squares scheme (DLS) which is used to damp the norm of joint velocity [2] [3]. The damping factor should be carefully chosen to guarantee good behaviors near singularities without affecting the normal working. With the Singular Value Decomposition (SVD) of the Jacobian matrix, this choice is also easy to understand. The intuitive idea is to apply a switch between the pseudoinverse far from the singularities and the damped solution near any singularity configuration using some comparative criteria of the damped factor and some threshold [5],[6],[8],[9],[10] and [11]. When only DLS scheme is used, there are several ways to find the optimal value of the damping factor. It has been chosen constant [7], and it is computed from the estimation of the smallest singular value [3]. There are relatively simple ways to compute this factor based on conditions imposed on the maximum allowable values of the joint velocity, cartesian velocity, condition number or by minimizing the residual function with respect to this factor [6],[12].

Other solutions are proposed to overcome the problem of perturbation of the good behavior. They consist of regularizing the smallest singular value, by omitting it [13] or by applying a filter [12]. However, the main drawback of these methods is the heavy computation time cost. In order to avoid this problem, the solution proposed in [12] adopts a recursive numerical algorithm which estimates both the smallest singular value and the associated singular vector. This estimate is used to achieve a solution in which the component associated with the smallest singular value is damped more than the others. An extension of this scheme has been proposed in [8] by estimating not only the smallest singular value but also the second smallest. From the estimation of more than one small singular value, this scheme is advised against due to its heavy computation time. Furthermore, the smallest estimated value is still only an approximation of the exact value. The selectively damped least square, from which this work is inspired, present an extension of [12] for all the singular vectors. Unlike of above scheme, the SDLS is based on the exact values of all components of SVD. In addition, the SDLS method analyzes each joint angle individually and decides how much this joint can move to reach the desired end-effector position, and then it compares this to the real distance. If there is an overrun of the real distance, then the motion in the direction of this joint angle must be damped more severely [11]. SDLS exhibits a better performance than normal DLS in tracking reachable target position rather than unreachable position without oscillation.

In the case of need, where there are several singular values equal to zero, all performed techniques even SDLS have no solutions. An example of this situation is the case of a robotic anthropomorphic hand with the index pointing out. Here, the finger is an outstretched chain and all joint angles are zero. So there is at least one zero singular value. Therefore, the aim of this work is to adapt the SDLS philosophy to the Jacobian transpose method which precludes the apparition of all kind of dividing operation.

In the literature of the inverse kinematics, few works have been realized using the Jacobian transpose method. The first contribution date back to 1984 [14], where the stability of the technique is proved by using the Lyapunov function and some assumptions. An augmented Jacobian transpose method has been proposed in [15] with the aim of taking into account the obstacle avoidance and limited joint range. It is well known that the Jacobian transpose method can be computed in few iteration, does not require the inverse Jacobian calculation but oscillates near the target. This undesirable behavior can be reduced by a damping factor such as the adopted in the SDLS scheme.

The goal of this paper is to extend the SDLS applied to the Jacobian inverse proposed in [11] to the case of the Jacobian transpose method. To reduce the computation time, a switching process has been done between the proposed technique and the original Jacobian transpose method.

This document is organized as follows. Section II gives a description of the original SDLS method and the modifications introduced to it. The inverse kinematics scheme based on the Jacobian transpose is reviewed in section III, while in section IV, the technique is used to define the proposed method MSDLS using the JT scheme. Section V is dedicated to present the simulated results of comparative studies between MSDLS and the switched JT with MSDLS applied to JT using the index finger of our experimental platform of the HANDLE project [16], *the Shadow hand*.

## II. INVERSE KINEMATICS BASED ON SELECTIVELY DAMPED LEAST SQUARES WITH OPTIMAL INCREMENT

By definition, the inverse kinematics problem consists of solving for a particular joint configuration in terms of given cartesian configuration. This problem requires solving a strong nonlinear system that can have some undetermined configurations, called singularities. The alternative approach to dealing with the problem of no-linearity is to approximate the system as a linear system and apply an iterative method through the Jacobian matrix. There are two configurations of the singularity, when the target position is reachable, the end-effector can not move in those directions in which the associated singular values equal or near to zero. Whereas, the second configuration is when the target position is unreachable. In both cases, a simple pseudoinverse method yields to instable system. Using SVD theory, the instability means the existence of zero singular value. The pseudoinverse method of the Jacobian matrix is defined by:

$$\dot{q} = J^\dagger \dot{x} \quad (1)$$

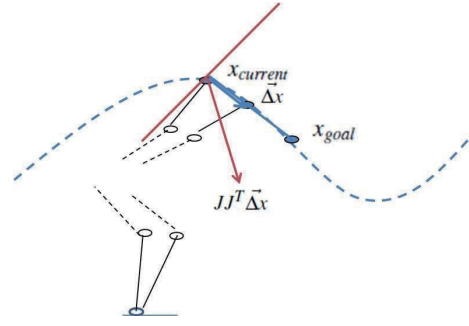


Fig. 1. Actual (dotted line) and linear approximation of the motion

Where  $J^\dagger$  is the pseudoinverse of the Jacobian matrix of the end-effector  $J$  ( $m \times n$ ). The SVD of  $J$  allows to describe the system (1) by two subspaces;  $U(m \times m)$  and  $V(n \times n)$  which are respectively composed by  $m$  output and  $n$  input singular vectors.

$$J = U\Sigma V \quad (2)$$

where the  $m \times n$  matrix  $\Sigma$  has the block matrix form:

$$\Sigma = \begin{bmatrix} S_{r \times r} & 0_{r \times (n-r)} \\ 0_{(m-r) \times r} & 0_{(m-r) \times (n-r)} \end{bmatrix}, \quad S = \text{diag}(\sigma_1, \dots, \sigma_r)$$

For convenience, the diagonal elements of  $r \times r$  matrix  $S$  can be ordered so that  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \geq 0$  where  $r = \text{rank}(J)$ . Hence, the equation (1) can be rewritten as follow:

$$\dot{q} = \sum_{i=1}^r \frac{1}{\sigma_i} v_i u_i^T \dot{x} \quad (3)$$

This leads to an iterative solution of inverse kinematics problem expressed in function of  $\Delta x$  and  $\Delta q$  the incrementing update of the end-effector position and joint angle respectively.

For a known  $x_{\text{current}}$  and  $x_{\text{goal}}$ , the update end-effector position given by Fig.1 is:

$$\Delta x = x_{\text{goal}} - x_{\text{current}} \quad (4)$$

This  $\Delta x$ , which should be quite small, corresponds to a  $\Delta q$  given by:

$$\Delta q = \sum_{i=1}^r \frac{1}{\sigma_i} v_i u_i^T \Delta x \quad (5)$$

Unlike the fixed choice of the maximum value of  $\Delta x_{\text{max}}$  in [10] and [11], there is an optimal choice of  $\Delta x_{\text{max}}$  based in the sensitivity of the solution of (1) as a function of the maximum allowable joint change  $\Delta q_{\text{max}}$  and condition number of the Jacobian matrix, denoted by  $\kappa$ , which is given by [4]:

$$\frac{\|\Delta q\|}{\|q\|} \leq \kappa(J) \frac{\|\Delta x\|}{\|x_{\text{current}}\|} \quad (6)$$

where  $\kappa(J) = \frac{\sigma_1(J)}{\sigma_r(J)}$ , so, in the extreme case when  $\Delta q = \Delta q_{\text{max}}$ , it is clear that:

$$\|\Delta x_{\text{max}}\| = \kappa(J) \frac{\|\Delta q_{\text{max}}\|}{\|q\|} \|x_{\text{current}}\| \quad (7)$$

When  $\|\Delta x_{max}\|$  is larger than the allowable physical change, the matrix  $J$  is ill-conditioned, this value should be restricted to that limit, which we denote as  $\|\Delta x_{allowable}\|$ . Using this updating of  $\Delta x_{max}$ , the modified SDLS is denoted as MSDLS.

Using the SVD theory, the SDLS scheme aims to control the effect of each input singular vector individually on the output singular vector. Buss and Kim in [11] proposed this technique to control rigid multibodies with multiple end-effectors.

It has been shown [11] that the SDLS scheme worked better than the DLS and the pseudoinverse methods in terms of tracking for both reachable and unreachable target position, but it is slow since the SDLS requires to treat the effect of each singular vector; as input and output, individually. However, the worst feature of this method is when the singular value is equal to or near zero. In this case, a truncated SVD has been done wherein all singular values smaller than a certain threshold are omitted [13].

### III. JACOBIAN TRANSPOSE AS INVERSE KINEMATICS METHOD

The Jacobian transpose method consists of replacing the pseudoinverse of  $J^\dagger$  with the transpose matrix  $J^T$  as:

$$\Delta q = \gamma \cdot J^T \Delta x \quad (8)$$

To ensure the convergence towards the target position Sciavicco and Siciliano [15] proposed the following coefficient:

$$\gamma = \alpha + (\Delta x^T \dot{x}_{goal})(\Delta x^T J J^T \Delta x)^{-1} \quad (9)$$

where  $\alpha > 0$ . With this choice, they proved that the dynamic error  $\Delta x$  decreases considerably to zero, and concluded that the second term of (9) can be dropped since with a purely proportional control law:

$$\Delta q = \alpha J^T \Delta x \quad (10)$$

the Lyapunov function stability based on the error dynamic holds under a certain condition of  $\|\dot{x}_{goal}\|$  [15]. The attractive motion towards the  $x_{goal}$  produced by the JT method given by (10) can be justified as virtual force because it seems as pulling the end-effector with an elastic force inspired from the relationship between the joint torque vector  $\tau$  and cartesian force vector  $F$  [14].

$$\tau = J^T F \quad (11)$$

The real change of the end-effector position  $x_{real}$  is obtained by using the forward kinematics of updated joint angle given in (10):

$$\Delta x_{real} = \alpha J J^T \cdot \Delta x \quad (12)$$

It is clear to see that for all  $J$ ,  $\Delta x$  and  $\alpha \geq 0$ , the following relation is kept:  $\langle J J^T \Delta x, \Delta x \rangle \geq 0$ . This means that  $\Delta x_{real}$  and  $\Delta x$  manifest in the same half plane (see Fig.1).

The convergence rate of the JT method depends essentially

on the scale  $\alpha$  which was chosen in [10] and [11] as the orthogonal projection norm of  $\Delta x$  on  $\Delta x_{real}$ :

$$\alpha = \frac{\langle \Delta x, J J^T \Delta x \rangle}{\langle J J^T \Delta x, J J^T \Delta x \rangle} \quad (13)$$

This choice ensures that  $\|\Delta x_{real}\|$  is as close as possible to  $\|\Delta x\|$ . The JT method as inverse kinematics is characterized by a few iteration process, which is fairly accurate near the desired target position but it manifest a strong shaking due to the direction changing of  $\Delta x$  by  $J^T$ . Considering the advantage of no-existence of singularities and low computation cost of the JT method, we have used the above filtering strategy to improve this method against the unwanted shaking behaviors. The idea is to apply the JT approach when the end-effector is far from the desired point, and the proposed MSDLS for transpose method to cancel the oscillations.

### IV. JACOBIAN TRANSPOSE AS MODIFIED SELECTIVELY DAMPED LEAST SQUARES METHOD

It is to be mentioned that, unlike the MSDLS (or all methods which require the inverse computing), the JT method admits a solution all the time. With this method, the solution cannot be reached with very high precision, which therefore deviate a little from the exact solution given by the pseudoinverse method, which may produce undesired behavior near the desired solution. However, like the MSDLS, it is possible to damp those solutions which exceed the desired position. In the following, we adopt an algorithm with the same steps as MSDLS with some adjustments to apply it on the JT method

According to the SVD theorem and equation (10), in this case the update angle change  $\Delta q$  is given by:

$$\Delta q = \alpha \sum_{i=1}^r \sigma_i v_i u_i^T \Delta x \quad (14)$$

If we note by  $\Delta x = \sum_{i=1}^r N_i \cdot u_i$  the representation of  $\Delta x$  in terms of the input singular vector, then  $N_i$  is the component of the input in  $u_i$  vector, where  $N_i = \langle \Delta x, u_i \rangle$ . As shown in [11], the proposed algorithm is based on the relative magnitude parameter  $\rho_j$  which allows to define the length of the current end-effector position  $\Delta x_{current}$  due to the small change of the  $j_{th}$  joint angle

$$\rho_j = \left\| \frac{\partial \Delta x_{current}}{\partial q_j} \right\| \quad (15)$$

On the other hand, according to equation (10), the response of input  $\Delta x$  acting on  $u_i$  would change the  $j_{th}$  joint angle, denoted by  $q_{ij}$ , as:

$$q_{ij} = \alpha N_i \sigma_i v_{ij} \quad (16)$$

where  $v_{ij}$  is the entry of  $V$  singular vector located in  $i_{th}$  column and  $j_{th}$  row. In return, the real magnitude of the end-effector position moved by  $q_{ij}$  can be calculated as:

$$M_i = \alpha N_i \sigma_i \sum_{j=1}^n \rho_j \cdot |v_{ij}| \quad (17)$$

Joint j	$a_j$	$d_j$	$\alpha_j$	$\theta_j$
1	0	0	$-\pi/2$	$\theta_1$
2	45	0	0	$\theta_2$
3	25	0	0	$\theta_3$
4	24	0	0	$\theta_4$

TABLE I  
D-H PARAMETERS FOR THE INDEX FINGER

If the value of  $M_i$  is greater than  $N_i$ , namely:

$$\alpha \sigma_i \sum_{j=1}^n \rho_j \cdot |v_{ij}| \geq 1 \quad (18)$$

a strong damping operation must be done in this direction to impede all kinds of excess in the desired distance to move by the end-effector. In [11], the damping factor is defined as the ratio  $\lambda_i = \frac{N_i}{M_i}$ . In this case, the ratio becomes quite small near singularities, since  $M_i \rightarrow \infty$  as  $\sigma_i \rightarrow 0$ . However, according to equation (17), the value of  $M$  always takes a finite number which leads to a weak damped factor when  $M$  becomes greater than  $N$ . Intuitively, to make this factor more effective, we have adopted an exponential form to this ratio:

$$\lambda_i = \begin{cases} 1 & \text{if } \frac{N_i}{M_i} < 1 \\ \left(\frac{N_i}{M_i}\right)^n & \text{otherwise} \end{cases} \quad (19)$$

When  $M_i \gg N_i$ , the ideal choice of  $n$  which allows higher cancelation in  $i^{th}$  singular vector direction has to be large. However, it has been seen; from several experimental implementations, that this factor leads to an undesirable behavior starting from a certain value. In the opposite case, when a light damping factor is required, the  $n$  should have a small value but not very small otherwise the proposed technique converge slowly. Therefore, an adequate choice of  $n$  is important to ensure this compromise. In order to summarize all the steps to achieve the Jacobian transpose with the MSDLS method for the inverse kinematics, we propose the implementation of the idea as shown in Algorithm 1.

## V. SIMULATED RESULTS

In this part, we adopt the calculation of the inverse kinematics for a redundant configuration. As indicated in the introduction, the method was run on the index finger of the Shadow hand [17]. The perspective work consist in implementing the obtained results to the rest of the fingers with the aim of reaching a grasping configuration. Here, each finger is considered as an independent manipulator with the base frames of all five fingers coinciding and attached to the palm of the robot hand. From the mechanical design of the Shadow hand, all Denavit-Hartenberg (D-H) parameters of the index finger are given in table I.

It is clear that, if we are interested only to the fingertip position  $x \in R^3$ , the finger is in a redundant configuration ( $q \in R^4$ ). One of the singular configurations of the index finger appears when all joint variables are set to zero. This configuration is both reachable and a singular point which will be our comparative study between the MSDLS

**Algorithm 1** JT method as MSDLS for single multibody chain

**Input:**  $\Delta x_{allowable}, \Delta q_{max}, \sigma_i, u_i, v_i, q, x_{current}, x_{goal}, \varepsilon, J$   
**Output:**  $\Delta q$

---

```

1:  $\kappa \leftarrow \frac{\sigma_1}{\sigma_r}$ 
2:  $\Delta x_{max} \leftarrow \kappa \frac{\|\Delta q_{max}\|}{\|q\|} \|x_{current}\|$ 
3: if  $\Delta x_{max} > \Delta x_{allowable}$  then
4:    $\Delta x_{max} \leftarrow \Delta x_{allowable}$ 
5: end if
6:  $\Delta x \leftarrow x_{goal} - x_{current}$ 
7: while  $\|\Delta x\| \geq \varepsilon$  do
8:   if  $\|\Delta x\| \geq \Delta x_{max}$  then
9:      $\Delta x \leftarrow \Delta x_{max} \frac{\Delta x}{\|\Delta x\|}$ 
10:  end if
11:   $\rho_j \leftarrow \left\| \left( \frac{\partial x_1}{\partial q_j}, \frac{\partial x_2}{\partial q_j}, \dots, \frac{\partial x_m}{\partial q_j} \right)^T \right\|$ 
12:   $\alpha \leftarrow \frac{\langle \Delta x, J J^T \Delta x \rangle}{\langle J J^T \Delta x, J J^T \Delta x \rangle}$ 
13:  for  $i = 1$  to  $r$  do
14:     $N_i \leftarrow \langle \Delta x, u_i \rangle$ 
15:     $M_i \leftarrow \alpha N_i \sigma_i \sum_{j=1}^n |v_{j,i}| \cdot \rho_j$ 
16:     $D_i \leftarrow \frac{N_i}{M_i}$ 
17:    if  $D_i \leq 1$  then
18:       $\gamma_i \leftarrow (D_i)^n \cdot \Delta q_{max}$ 
19:    else
20:       $\gamma_i \leftarrow \Delta q_{max}$ 
21:    end if
22:     $\varphi_i \leftarrow \sigma_i N_i v_i$ 
23:     $\eta_i \leftarrow \|\varphi_i\|_\infty$ 
24:    if  $\eta_i \geq \gamma_i$  then
25:       $\varphi_i \leftarrow \gamma_i \frac{\varphi_i}{\eta_i}$ 
26:    end if
27:  end for
28:   $\Delta q \leftarrow \sum_{j=1}^r \varphi_j$ 
29:  if  $\|\Delta q\| \geq \Delta q_{max}$  then
30:     $\Delta q \leftarrow \Delta q_{max} \cdot \frac{\Delta q}{\|\Delta q\|}$ 
31:  end if
32:   $x_{current} \leftarrow x_{current} + J \cdot \Delta q$ 
33: end while
34: return  $\Delta q$ 

```

---

and switched JT with MSDLS applied to JT in term of computation time and accuracy. The index finger inverse kinematics calculation was performed from an initial configuration distanced  $\|\Delta x\| = 181.5291mm$  from the goal point. According to the clamping function [11] defined in the lines 8 and 9 of algorithm 1, the generated trajectory is presented by a straight line with a maximum increment length  $\Delta x_{max}$ . The choice of this length has an important role to decide the performance of the tracking task. In some applications this increment may be important to reach a fast task. However, in others, when the system has less information about the object to manipulate, the increment length is required to be small in order to achieve a slow and accurate task. This kind of calculation is required in the grasping operations and in the *in-hand manipulation* of objects [16] especially for fragile



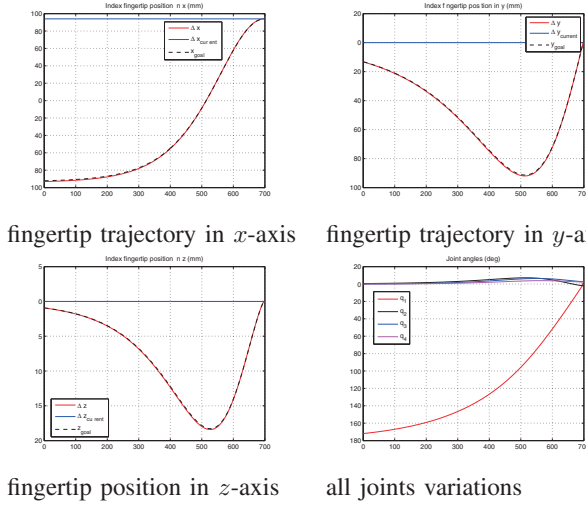


Fig. 2. Positions and angles obtained results using the switched method with  $\Delta x_{max} = 1mm$

objects. This problem can also be resolved by adopting the characteristics of the force sensing if the hand is equipped with such capabilities.

The results shown in Fig.2 and Fig.3 present the application of the switched JT with MSDLS applied to JT and MSDLS method, respectively. The initial configuration for both approach was:  $q = (-3 \ 0 \ 0 \ 0)^T rad$ . In both cases, the positions and joints trajectories are similar and take almost the same iteration number to reach the desired position ( $i \simeq 700$ ). However, this similarity does not apply in term of computation time. Since the singular value decomposition is a computationally expensive operation, the MSDLS method seems to be slower but it gives a more precise solution when compared to the Jacobian transpose method which does not need SVD computation. In the switched case, when the error between the current and the goal position is less than the threshold ( $\epsilon \simeq 0.08$ ), while the MSDLS applied to JT has almost the same computation time as MSDLS.

The joint variations in Fig.3 reveal also that the generated trajectory is at singularity configuration all along the simulation time. This is evident through the handled zero value taken by  $(q_2, q_3, q_4)^T$  and only  $q_1$  changes, which produces a planar arc trajectory. There are several factors that affect the computation time, of which the most important are; the type of trajectory, the maximum increment, the threshold value which gives the convergence and the calculation process end and obviously, the hardware on which the implementation is executed.

The inverse kinematics and all required operations were computed and simulated in MATLAB, running on a laptop with a Duo Core Processor at 2.53 GHz.

The results shown in Fig.4 demonstrate that for small maximum increments the proposed method takes less time to reach the goal position than the MSDLS method. From  $\Delta x_{max} \simeq 15mm$ , this difference becomes less significant. This part is zoomed in to show more clearly the improve-

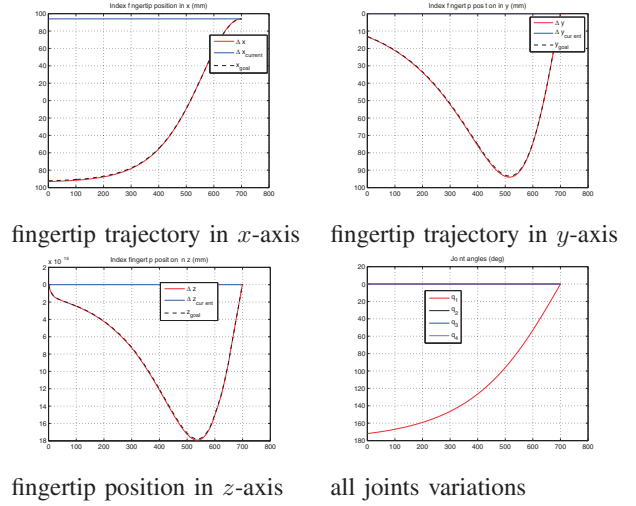


Fig. 3. Positions and angles obtained results using the MSDLS method with  $\Delta x_{max} = 1mm$

ment, using the computation ratio between the execution times of the two methods (Fig.5). The ratio in percent is defined as:  $\rho = \frac{t_{original} - t_{proposed}}{t_{original}} 100$ . Here, the curve can be observed as two regions. In the positive region, the proposed method provides less computation times until the maximum increment  $\Delta x_{max} \simeq 15mm$  where the curve starts taking negative sign. In this region, the MSDLS approach becomes faster than the proposed method. On the other hand, the inflexion point is not constant and depends on the length of the trajectory and the threshold of convergence. During in-hand manipulation tasks, a large increment length is not advisable, in particular when performing the rolling and sliding of the finger over the object surface. Therefore, this proposed method is suited to the application by using smaller increments. Fig.6 shows the impact of the threshold of convergence on the computation time repeating the above simulation but now with the fixed maximum increment  $\Delta x_{max} = 0.1mm$ .

Like the above figure, The Y-axis is computed by the ratio between the computation time of MSDLS and the proposed method for different convergence thresholds. The negative part of the curve indicates that the MSDLS is very accurate and can give for a determined time a quite insignificant convergence error. This result is well expected, but starting from  $\epsilon = 0.08 \text{ mm}$ , the proposed method exhibits faster execution, and the time ratio varies around 10%. In robotic hand applications where the calculation of the inverse kinematics of the five fingers is required this 10% translates in an important laps time improvement. This means that the proposed method reaches a neighborhood of the goal position in short time but with moderate accuracy. Moreover, in robotic in-hand manipulation, this cannot be considered as drawback since  $0.1mm$  is already considered a very good precision and due to the soft-contact characteristic the contact is defined as small area rather than a point.

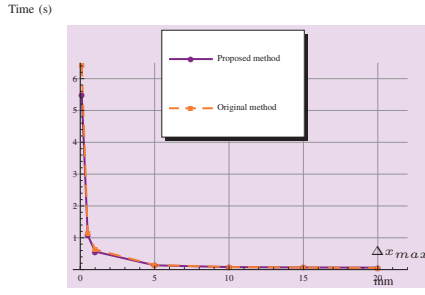


Fig. 4. Computation times for two methods for convergence threshold  $\epsilon = 0.08$

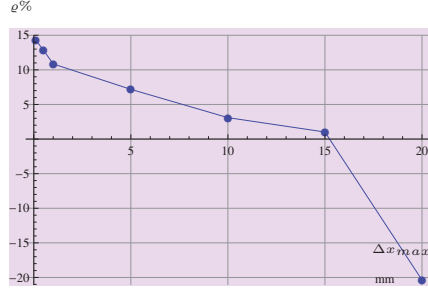


Fig. 5. Computation time ratio between the two methods for convergence threshold  $\epsilon = 0.08$

## VI. CONCLUSION

In this work, the inverse kinematics for the index finger of an anthropomorphic robotic hand is investigated. This finger exhibits a redundancy configuration. The recently developed Selectively Damped Least Squares scheme has been modified by applying an optimal increment instead of a fixed one. Using this modification, we proposed a method which consists of extending MSDLS to cover the Jacobian transpose method, thus avoiding all kind of singularity configuration since the JT method does not involve any division operation.

The short computation time yield by the JT method is another motivation of our choice. However, this method exhibits an undesirable shattering behavior near the goal position. Taking inspiration of the MSDLS advantage, this undesirable behavior can be damped. Therefore, a combined method was developed to adopt a switching between the JT and the MSDLS applied to the JT to reduce the computation and improve the convergence error. With regard to computation time, it is shown that in the case of small increments, the proposed method runs in a shorter time with respect to the MSDLS method. This difference can be much more noteworthy when all fingers are employed for hand manipulation tasks, especially to achieve rolling and sliding in-hand manipulation. The convergence threshold parameter is one of important factors which affect the efficiency of the proposed method. The obtained results show that the proposed method exhibits less computation time from a certain convergence threshold than the MSDLS method. In soft-contact hand manipulation, where the contact is an area rather than a point, the required accuracy is not very high and our method can fulfill the task efficiently. With these arguments, the use of our proposed method is justified and

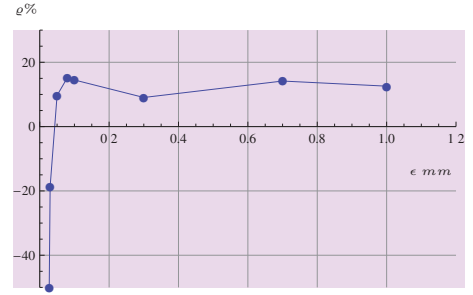


Fig. 6. Impact of the convergence threshold on computation times for two methods with  $\Delta x_{max} = 0.1mm$

since it yields better results than the original SDLS method, a version in the C++ language is being developed and will cover the complete model of the anthropomorphic hand used in our lab as an experimental platform for dexterous manipulation.

## REFERENCES

- [1] Y. Li and I. Koa. A review of Modelling of Soft-contact fingers and stiffness control for dexterous manipulation in robotics. *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, Seoul, Korea- May 21-26, 2001.
- [2] L.M. Phuoc, P. Martinet, S. Lee and H. Kim. Damped least square based genetic algorithm with Gaussian distribution of damping factor for singularity-robust inverse kinematics. In *Journal of Mechanical Science and Technology* 22, 1330-1338, 2008.
- [3] O. Egeland, J.R. Sagli, I. Spangelo and S. Chiaverini. A Damped Least Squares Solution to Redundancy Resolution. In *Proceeding of 1991 IEEE Conference of robotics and Automation*, pages 945-950, Sacramento, CA, April 1991.
- [4] A. Ben-Israel and T.N.E. Graville. *Generalized Inverses: Theory and Applications*. New York, Inc: Springer-Verlag, 2003.
- [5] A.A. Maciejewski, C.A. Klein. The Singular Value Decomposition: Computation and Applications to Robotics. In *The international Journal of Robotics Research*, vol. 8, No. 6, Decembre 1989.
- [6] A.S. Deo, I.D. Walker. Robust Subtask performance with Singularity robustness using Optimal Damped Least-Squares. In *Proceeding of the 1992 IEEE International Conference of robotics and Automation*. Nice, France, May 1992.
- [7] C.W. Wampler. Manipulation Inverse Kinematic Solutions Based on Vector Formulations and Damped Least-Squares Methods. In *IEEE Trans of Systems, Man, and Cybernetics*, Vol. SMC-16, No.1 January/February 1986.
- [8] S. Chiaverini, B. Siciliano and O. Egeland. Review of Damped Least-Squares Inverse Kinematics with Experiments on an Industrial Robot Manipulator. In *Trans on Control System Technology*, Vol. 2, No. 2, June, 1994.
- [9] L.M. Phuoc, P. Martinet, S. Lee and H. Kim. Damped Least-Square based genetic algorithm with Gaussian distribution of damping factor for singularity robust inverse kinematics. In *Journal of Mechanical Science and Technology* 22. 1330-1338. 2008.
- [10] S.R. Buss, Introduction to Inverse Kinematics with Jacobian transpose, Pseudoinverse and Damped Least Squares Method. Typeset manuscript, available from <http://math.ucsd.edu/~sbuss/ResearchWeb>, April 2004.
- [11] S.R. buss and J.S. Kim, Selectively Damped Least Squares for Inverse Kinematics. In *Journal of Graphics Tools*, 37-49, 2005.
- [12] R.V. Patel and F. Shadpey. *Control of Redundant Robot Manipulators, Theory and Experiments*. Springer Berlin Heidelberg New York, 2005.
- [13] P.C. Hansen. The truncated SVD as a method for regularization. *Numerical Analysis Project Manuscript*. NA-86-36. October 1986.
- [14] W.A. Wolovich and H. Elliott. A computation technique for inverse kinematics. *Proceeding of 23rd Conference on Decision and Control*. Las Vegas. NV December 1984.
- [15] L. Sciacivco and B. Siciliano. A Solution Algorithm to the Inverse Kinematics Problem For Redundant Manipulators. *IEEE Journal of Robotics and Automation*. Vol4, No. 4. August 1988.
- [16] <http://www.handle-project.eu>.
- [17] <http://www.shadowrobot.com>.